IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s):	Messaoud Benantar		
Assignee:	International Business Machines Corporation		
Title:	Method and System for Computing Digital Certificate Trust Paths Using Transitive Closures		
Serial No.:	10/045,112	Filing Date:	January 10, 2002
Examiner:	Shin Hon Chen	Group Art Unit:	2131
Docket No.:	AUS920010943US1	Customer No.	65362

Austin, Texas
August 27, 2008

FILED ELECTRONICALLY

APPEAL BRIEF UNDER 37 CFR § 41.37

Dear Sir:

Applicant submits this Appeal Brief pursuant to the Notice of Appeal filed in this case on May 27, 2008. The fee for this Appeal Brief is being paid electronically via the USPTO EFS. The Board is authorized to deduct any other amounts required for this appeal brief and to credit any amounts overpaid to Deposit Account No. 090447.

I. REAL PARTY IN INTEREST - 37 CFR § 41.37(c)(1)(i)

The real party in interest is the assignee, International Business Machines Corporation, as named in the caption above and as evidenced by the assignment set forth at Reel 012501, Frame 0659.

II. RELATED APPEALS AND INTERFERENCES - 37 CFR § 41.37(c)(1)(ii)

Based on information and belief, there are no appeals or interferences that could directly affect or be directly affected by or have a bearing on the decision by the Board of Patent Appeals and Interferences in the pending appeal. Pursuant to current Patent Office practice, Appendix "A" contains copies of all decisions rendered by a court or the Board in this "Related Appeals and Interferences" section, and is intentionally provided as an empty appendix.

III. STATUS OF CLAIMS - 37 CFR § 41.37(c)(1)(iii)

Claims 1-30 are pending in the application. Claims 1-30 stand rejected. The rejection of claims 1-30 is appealed. Appendix "B" contains the full set of pending claims.

IV. STATUS OF AMENDMENTS - 37 CFR § 41.37(c)(1)(iv)

On July 28, 2008, Applicant filed a response to the Final Office Action, but did not make any amendments.

V. SUMMARY OF CLAIMED SUBJECT MATTER - 37 CFR § 41.37(c)(1)(v)

The claims of the present patent application are directed to a method, system, apparatus, and computer program product are presented for managing digital certificates. When entities need to engage in a secure transaction or open a secure communication link, they may exchange digital certificates in order to provide a public key or reference information to a public key for the opposing entity, thereby requiring validation of a received certificate. Rather than construct a trust path for each validation event, hierarchical certifications and peer-to-peer cross-certifications among a set of certificate authorities are represented by a set of trust relations, and trust path information is generated using a "transitive closure computation" and an "all pairs shortest paths" computation over the set of trust relations and then incrementally updated as the set of trust relations changes. Computations related to trust paths can be delegated to a central agent in a trust web.

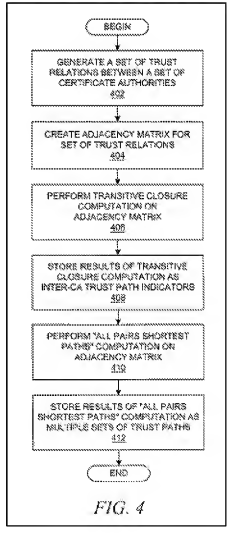

FIG. 4

The subject matter defined in independent claims 1, 4, and 7 may be understood with reference to the example embodiments depicted in Figures 4 and 5 which depict a method, apparatus and computer program product for processing digital certificates within a data processing system. Referring now to Figure 4, the method begins at step 402 when a set of trust relations are determined between a set of certificate authorities (CAs) in a trust web. At step 404, the set of trust relations are represented in an adjacency matrix in which each cell in the adjacency matrix corresponds to a pair of certificate authorities. At step 406 (and Figure 5), a "transitive closure computation" is performed on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities. And at step 410, a separate "all-pairs-shortest-paths" computation is performed on the adjacency matrix to

generate multiple sets of shortest trust paths between the certificate authorities. *See, e.g.*, Application, Figures 4, 5A, and ¶¶ 67-76.

The subject matter defined in independent claims 10, 14, and 18 may be understood with reference to the example embodiments depicted in Figures 6 and 8B-C which depict a method, apparatus and computer program product for operating certificate authorities within a data processing system. Referring now to Figure 8B, the method begins at step 822 when a trust relation with a second certificate authority is established at a first certificate authority (CA). At step 824, a trust relation update message is sent to a central trust web agent which processes the trust relation information for a set of certificate authorities within a trust web. *See, e.g.*, Application, Figures 6, 8B-C, and ¶¶ 87-88 and 94-98.

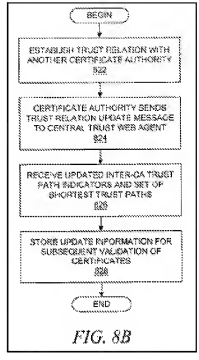

FIG. 8B

Finally, the subject matter defined in independent claims 22, 25, and 28 may be understood with reference to the example embodiments depicted in Figures 6 and 8B-C which depict a method, apparatus and computer program product for operating certificate authorities within a data processing system. Referring now to Figure 8C, the method begins at step 832 where a central trust web agent receives a trust relation update message from a certificate authority (CA) and processes the trust relation information for a set of certificate authorities within a trust web, where the trust relation update message indicates a change in a set of trust relations for the certificate authority. At step 834, a set of trust relations for the set of certificate authorities within the trust web is modified based on an indicated request in the trust relation update message. *See, e.g.*, Application, Figures 6, 8B-C, and ¶¶ 87-88 and 94-98. As seen from the foregoing, the subject matter of the independent claims is set forth in the Application at Figures 4, 5A-5B, 6-7, and 8A-8C and the associated description, including ¶¶ 12, 13-15, 18-32, and 66-98. While Applicants have identified passages from the specification to explain the independent claim subject matter and how it may be

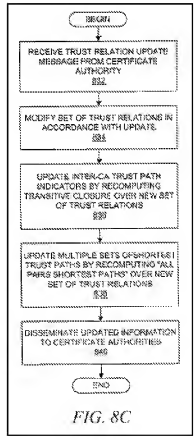

FIG. 8C

implemented with a computer program product in a data processing system in a network, it will be appreciated that the referenced description includes contextual information to provide an overall context for an example embodiments, and therefore should not be used to improperly read limitations from the specification into the claims.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

In the Final Office Action dated May 27, 1008, claims 1-30 were rejected as anticipated by U.S. Patent No. 6,134,550 to Van Oorschot et al. ("Oorschot").

VII. ARGUMENTS

A. The "Transitive Closure Computation" Requirement Variously Recited In Claims 1-9, 24, 27, and 30 Is Not Anticipated by Oorschot

In response to the Examiner's original rejection of claims 1-9 as being anticipated by Van Oorschot, Applicant explained that Van Oorschot's disclosed system (for employing trusted paths to determine the validity of a certificate) does not anticipate the present invention's scheme for computing digital certificate trust paths by, *inter alia*, "performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities." *See, e.g.*, claims 1, 24, 27, and 30. In response, the Examiner makes the wholly unsupported assertion that "the examiner disagrees. Oorschot reference discloses computing shortest path of all certificate authority in transitive adjacency matrix. (Oorschot: column 4, lines 64-67 and Figures 7a and 7b)." *See*, Final Office Action, p. 6. In particular, the Examiner asserts that both the "transitive closure computation" and the "all-pairs-shortest-paths" computations were met by the same passage from Oorschot, namely col. 4, lines 52-57. *See*, Final Office Action, p. 2. In the Response to Final Office Action, Applicant explained that the "transitive closure computation" requirement is *separate* from the all-pairs-shortest-paths computation and is not disclosed as a separate requirement by Oorschot. In the responsive Advisory Action, the Examiner asserted for the first time that Oorschot's disclosure of a "compilation of certificate chain data to generate a table of trusted relationships among the certificate issuing units (VO: column 4 lines 52-62)" meets the claimed "transitive closure computation" requirement, and that "the compilation of certificate chain data is different from the shortest-path computation (VO: column 4 lines 65-67) in which the compilation of certificate chain data takes place before the shortest-path computation to ensure validity of path." *See,* Advisory Action, p. 3. With all due respect, the shift in the Examiner's latest rejection analysis to cite different passages from Oorschot appears

to be improperly driven in hindsight by Applicant's explanation, not by any legitimate reading of Oorschot's disclosure. Indeed, when the *entirety* of the cited Oorschot passage is considered, it becomes apparent that the "shortest trusted path" is an example of what is generated when Oorschot's certificate chain data is compiled, so there is no separate "transitive closure computation" disclosed by Oorschot:

> For example, where a high degree of compilation is performed, the certificate chain data may be a list of all certification authorities in a shortest trusted path starting with a subscriber's own CA and ending with the target CA that issued the certificate of the subscriber that sent a digitally signed message. The compiled certification authority trust data serves as certificate chain data that may be for example, a table of trust relationships among the certificate issuing units in a community of interest, to facilitate rapid validity determination of the certificate by a plurality of requesting units. By way of example, the compilation may consist of populating a database that can be repeatedly queried by multiple subscribers to provide a preferred chain of certificates in a shortest trusted path among two subscribers, or between their respective CAs. If preferred, the stored certificate chain data can also include the associated certificates, or other information such as revocation status information related to the associated certificates, for each of the certificate issuing units listed in the table.

Oorschot, col. 4, lines 52-67 (emphasis added). As the passage shows, the referenced compilation of certificate chain data from Oorschot is not separate from the "all-pairs-shortest-paths" computation, and therefore does not anticipate the separately claimed requirement of "performing a transitive closure computation." However, as seen in claim 1's separate recitation of the "transitive closure" and "all-pairs-shortest-path" computations, the recited "transitive closure computation" is distinct from the "all-pairs-short-path" computation, and is used to immediately determine whether a trust path exists between two certificate authorities before the actual path is determined:

> [0070] The **transitive closure** is then computed over the adjacency matrix (step 406). The **transitive closure** represents whether there is a path, i.e. set of edges, through the directed graph for any two nodes in the directed graph. Hence, the **transitive closure** can also be represented with a matrix with each cell reflecting whether or not there is a path between the entities that correspond to the row and the column for the cell.
>
> [0071] Several algorithms exist for computing the **transitive closure** matrix from an adjacency matrix; "Warshall's algorithm" is frequently used for this type of computation, which is also known as solving the reachability problem for a set of nodes in a graph.
>
> [0072] The present invention recognizes that a **transitive closure** computation can be applied to a trust web. The output of a **transitive closure** computation represents whether or not an established trust path exists between two certificate authorities that are involved in a certificate validation process; this output information may be termed "inter-CA trust

path indicator information" as it quickly indicates whether or not a trust path exists between two certificate authorities. The result of the **transitive closure** computation is then stored in an appropriate format (step 408), e.g., a simple file containing the matrix, one or more database records, or some other format.

[0073] An "all pairs shortest paths" computation is then performed on the adjacency matrix (step 410). The shortest paths that are discovered during the "all pairs shortest paths" computation are then stored in an appropriate format (step 412), e.g., a simple file containing a set of paths, a set of files containing a vector representing a path, a set of linked list data structures, a set of one or more database records, or some other format. The "all pairs shortest paths" computation is described in more detail below.

[0074] With reference now to FIGS. 5A-5B, the **transitive closure** computation and the "all pairs shortest paths" computation are depicted using an adjacency matrix that is used as input to the computations and the resulting matrices that are output from the computations.

[0075] In one embodiment of the present invention, an adjacency matrix might represent only the existence of relations between nodes in a directed graph. In other words, a simple adjacency matrix might have cells in which a value of "0" in a cell represents the non-existence of a relation between the corresponding nodes for the cell and a value of "1" represents the existence of a relation. In the present invention, the cells along the diagonals of this type of adjacency matrix are filled with zeroes because a trust relation between a certificate authority and itself is not represented, although in the realm of digital certificates, a certificate authority can be viewed as "self-certifying".

[0076] Referring to FIG. 5A, adjacency matrix 502 is depicted for the set of certificate authorities that are shown in FIG. 3B. Using this adjacency matrix, the **transitive closure** computation produces an output matrix in which each cell reflects whether or not there is a path between the entities that correspond to the row and the column for the cell. With respect to the present invention, output matrix 504 from the **transitive closure** computation represents a set of inter-CA trust path indicators that reflect whether or not there is a path between the corresponding certificate authorities.

See, Application, paragraphs 70-76 (emphasis added). As seen from the foregoing, the recited "transitive closure computation" is distinct from the "all-pairs-short-path" computation, and is used to determine, before the actual path is determined, whether there is a path through the directed graph for any two certificate authorities in the directed graph. Applicant submits further that those skilled in the art would understand that the transitive closure algorithm differs from the shortest path algorithm in that Oorschot requires storing the pairs of shortest paths, while the transitive closure calculations are simpler since they only deal with a true or false answer (where true means there is a path between two nodes and false otherwise) so that the transitive closure

only needs to store the boolean true or false for each given pair of CA certificates (0, or 1) that can be minimized to the bit-wise level.

While Applicant has distinctly recited the claim requirement of "performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities," the Final Office Action omits any explanation of how Van Oorschot anticipates this claim requirement since the cited passage (Van Oorschot, col. 4, lines 52-57) clearly makes no reference to such a "transitive closure computation." Accordingly, Applicant respectfully submits that this omission amounts to a failure to articulate a *prima facie* anticipation showing that each and every element of the claimed invention, arranged as required by claims 1-9, 24, 27, and 30, are found in the Oorschot reference, either expressly or under the principles of inherency. *See generally*, In re King, 801 F.2d 1324, 1326, 231 USPQ 136, 138 (Fed. Cir. 1986); Lindemann Maschinenfabrik GMBH v. American Hoist and Derrick, 730 F.2d 1452, 1458, 221 USPQ 481, 485 (Fed. Cir. 1984). Because of at least these differences between Oorschot and claims 1-9, 24, 27, and 30, Applicant requests that the anticipation rejection of claims 1-9, 24, 27, and 30 be withdrawn and the claims be allowed.

B. The Requirement That A Certificate Authority Send "A Trust Relation Update Message To A Central Trust Web Agent" Variously Recited In Claims 10-30 Is Not Anticipated by Oorschot

In response to the anticipation rejection of claims 10-30, Applicant respectfully submits that Oorschot's disclosed system (for employing trusted paths to determine the validity of a certificate) does not anticipate the present invention's scheme for computing digital certificate trust paths by, *inter alia*, having a certificate authority send "a trust relation update message to a central trust web agent, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a trust web," as variously recited in claims 10-30. Nor does Oorschot disclose the requirement variously recited in claims 22-30 of a central trust web agent which uses a "trust relation update message" from a certificate authority to modify a set of trust relations for the set of certificate authorities within the trust web. *See, e.g.*, claim 22. In finally rejecting claims 10-30, the Examiner cites Oorschot's disclosure (col. 5, lines 53-61, col. 6, lines 1-11, and col. 7, line 62 to col. 8, line 13) that the certificate chain data 209 and database 208 can be "periodically updated." *See*, Final Office Action, pp. 3-4. In response to Applicant's explanation that the claim requirements for the "trust relation update message" are not met by

Oorschot, the Examiner now asserts that Oorschot (col. 5, lines 53-61, col. 6, lines 1-11) discloses that "the CAs respectively provide certificate chain information to the central web agent for compilation and periodically provide update to the central agent in order to establish the up-to-date certificate chain data." *See,* Advisory Action, p. 3. However, as seen from the cited Oorschot passage which is set forth below, there is simply no reference in Oorschot of having a certificate authority send a "trust relation update message" to a central trust web agent, much less using the received "trust relation update message" to modify the set of trust relations at the central trust web agent:

> The certificate chain data 209 is compiled from certification authority trust data. Certification authority trust data may include for example, cross-certification data, revocation data and/or other data stored in a distributed directory (e.g., X.500 directories or LDAP-compliant repositories). The certificate chain data 209 is prepared once, and periodically updated as needed, for more than one subscriber and may be repeatedly used each time validation needs to occur.
>
> * * *
>
> .... The certificate validating unit 204 sends a query in the form of a request signal to the certificate chain constructing unit 206 to obtain certificate chain information relating to a preferred certificate chain between certificate issuing units in a trusted path in a community of interest. The certificate chain constructing unit 206 analyzes the certificate chain data 209 stored in the certificate chain data storage medium 208 and transmits certificate chain information to the certificate validation unit 204 to allow the certificate validation unit to determine the validity of the certificate to be validated in a rapid fashion.
>
> The certificate chain database 208 may be periodically updated, for example by the certificate chain data generator 400 periodically polling the distributed directory 302 or other sources of certificate data to determine whether updates in the certificate trust data has occurred (additional certificates, revocation of certificates etc.) and recompiling the necessary database entries. For example, if a certification authority (CA) is added to the community of interest, the certificate chain data generator 400 obtains the new information from the directory 302 and adds any links based on the certification authority trust data associated with that new certificate issuing unit to incorporate the trust relationship as certificate chain data 209 in the certificate chain data database 208. Therefore where a database 302 includes certificates indicating cross-certification among certificate issuing units, the certificate chain data generator 400 uses the cross-certification information to note a trust path between the corresponding certificate issuing units.

Oorschot, col. 5, lines 53-61, col. 6, lines 1-11, and col. 7, line 62 to col. 8, line 13 (emphasis added). Indeed, Oorschot discloses a variety of techniques for updating the certificate chain database 208 (including periodically polling the distributed directory or other sources of

certificate data), but conspicuously fails to disclose using trust relation update messages from the certificate authorities. In contrast, Applicant has distinctly claimed and described the role of the "trust relation update message" in "modifying a set of trust relations for the certificate authorities":

> [0095] The certificate authority then sends a trust relation update message to the central trust web agent (step 824); in a cross-certification operation, each certificate authority would be responsible for sending such a message to the central trust web agent. In response from the central trust web agent at some later point in time, the certificate authority would receive updated adjacency information, updated inter-CA trust path indicators (updated transitive closure information), and an updated set of shortest trust paths for the trust web (step 826). The certificate authority then stores this information for later use in a certificate validation procedure (step 828), and the process is complete. In this manner, the central trust web agent assumes the responsibility for performing transitive closure computations and "all pairs shortest paths" computations for the trust web.
>
> [0096] It should be noted that a certificate authority could receive updated information from the central trust web agent in response to an update from another certificate authority, thereby causing it to perform steps 826 and 828 without performing steps 822 and 824. This scenario could occur because, even though the certificate authority that receives this information does not have a direct trust relation with the certificate authority that caused the update, the receiving certificate authority could have a trust path with a certificate authority that established a new trust relation, which thereby affects the trust paths of the receiving certificate authority. Most certificate authorities should not be affected by a new trust relation, i.e. the effect should be localized, but it is possible that the effect could propagate throughout the trust web. This process is independent of any particular protocol that is used by the certificate authorities for establishing a new trust relation. In addition, this process is independent of any particular protocol or message format that is used to communicate information between the certificate authorities and the central trust web agent.
>
> [0097] With reference to FIG. 8C, a flowchart depicts a process by which a central trust web agent generates and disseminates trust web information to certificate authorities within a trust web. The process begins when the trust web agent receives a trust relation update message from a certificate authority (step 832), and the trust relation within the receive message is added or deleted from the current set of trust relations that is maintained by the central trust web agent (step 834). The trust web agent then performs the transitive closure computation (step 836) and also the "all pairs shortest paths" computation (step 838), after which it can store the information for later use. By comparing the newly generated transitive closure information and shortest path information with the previously generated information, the central trust web agent can determine which certificate authorities have been affected by the most recent trust relation update. Hence, the central trust web agent can communicate the appropriate

updated information to the affected certificate authorities (step 840), thereby completing the process from the perspective of the central trust web agent.

See, Application, paragraphs 95-97 (emphasis added).

While Applicant has distinctly recited the requirement of a "trust relation update message" that is sent to or received by a central trust web agent which "processes trust relation information for a set of certificate authorities within a trust web" in claims 10-30, the Final Office Action omits any explanation of how Oorschot anticipates this claim requirement. Nor does the Final Office Action explain how Oorschot discloses the requirement of "modifying a set of trust relations for the set of certificate authorities within the trust web based on an indicated request in the trust relation update message" recited in claims 24-30. Accordingly, Applicant respectfully submits that this omission amounts to a failure to articulate a *prima facie* anticipation showing that each and every element of the claimed invention, arranged as required by claims 10-30, are found in the Oorschot reference, either expressly or under the principles of inherency. *See generally*, In re King, 801 F.2d 1324, 1326, 231 USPQ 136, 138 (Fed. Cir. 1986); Lindemann Maschinenfabrik GMBH v. American Hoist and Derrick, 730 F.2d 1452, 1458, 221 USPQ 481, 485 (Fed. Cir. 1984). Because of at least these differences between Oorschot and the claims, Applicant requests that the anticipation rejection of claims 10-30 be withdrawn and the claims be allowed.

VIII. CLAIMS APPENDIX - 37 CFR § 41.37(c)(1)(viii)

A copy of the pending claims involved in the appeal is attached as Appendix "B."

IX. EVIDENCE APPENDIX - 37 CFR § 41.37(c)(1)(ix)

None.

X. RELATED PROCEEDINGS APPENDIX - 37 CFR § 41.37(c)(1)(x)

There are no related proceedings.

XI. CONCLUSION

A *prima facie* case of anticipation has not been established because the cited art fails to disclose Applicant's claimed scheme for computing digital certificate trust paths by "performing a transitive closure computation on the adjacency matrix …" that is separate and apart from the step of "performing an all-pairs-shortest-paths computation," as variously recited in claims 1-9, 24, 27, and 30. Nor does the cited art disclose Applicant's scheme for having a certificate authority send "a trust relation update message to a central trust web agent, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a

trust web," as variously recited in claims 10-30. Accordingly, it is respectfully urged that the rejection of the claims should not be sustained.

CERTIFICATE OF TRANSMISSION

I hereby certify that on October 27, 2008 this correspondence is being transmitted via the U.S. Patent & Trademark Office's electronic filing system.

/Michael Rocco Cannatti/

Respectfully submitted,

/Michael Rocco Cannatti/

Michael Rocco Cannatti
Attorney for Applicant(s)
Reg. No. 34,791

APPENDIX A - RELATED APPEALS AND INTERFERENCES

There are no decisions rendered by a court or the Board in any related proceeding.

APPENDIX B - PENDING CLAIMS

1. (Original) A method for processing digital certificates within a data processing system, the method comprising:

determining a set of trust relations between a set of certificate authorities (CAs) in a trust web;

representing the set of trust relations in an adjacency matrix, wherein a cell in the adjacency matrix corresponds to a pair of certificate authorities;

performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities; and

performing an all-pairs-shortest-paths computation on the adjacency matrix to generate multiple sets of shortest trust paths between the certificate authorities.

2. (Original) The method of claim 1 further comprising:

initiating a secure communication with a requester;

receiving a digital certificate for the requester; and

validating the digital certificate in accordance with an inter-CA trust path indicator and/or a shortest trust path.

3. (Original) The method of claim 2 wherein the digital certificate is formatted according to X.509 standards.

4. (Original) An apparatus for processing digital certificates within a data processing system, the apparatus comprising:

means for determining a set of trust relations between a set of certificate authorities (CAs) in a trust web;

means for representing the set of trust relations in an adjacency matrix, wherein a cell in the adjacency matrix corresponds to a pair of certificate authorities;

means for performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities; and

means for performing an all-pairs-shortest-paths computation on the adjacency matrix to generate multiple sets of shortest trust paths between the certificate authorities.

5. (Original) The apparatus of claim 4 further comprising:

means for initiating a secure communication with a requester;

means for receiving a digital certificate for the requester; and

means for validating the digital certificate in accordance with an inter-CA trust path indicator and/or a shortest trust path.

6. (Original) The apparatus of claim 5 wherein the digital certificate is formatted according to X.509 standards.

7. (Original) A computer program product in a computer-readable medium for use in a data processing system for processing digital certificates, the computer program product comprising:

instructions for determining a set of trust relations between a set of certificate authorities (CAs) in a trust web;

instructions for representing the set of trust relations in an adjacency matrix, wherein a cell in the adjacency matrix corresponds to a pair of certificate authorities;

instructions for performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities; and

instructions for performing an all-pairs-shortest-paths computation on the adjacency matrix to generate multiple sets of shortest trust paths between the certificate authorities.

8. (Original) The computer program product of claim 7 further comprising:

instructions for initiating a secure communication with a requester;

instructions for receiving a digital certificate for the requester; and

instructions for validating the digital certificate in accordance with an inter-CA trust path indicator and/or a shortest trust path.

9. (Original) The computer program product of claim 8 wherein the digital certificate is formatted according to X.509 standards

10. (Original) A method for operating certificate authorities within a data processing system, the method comprising:

establishing at a first certificate authority (CA) a trust relation with a second certificate authority; and

sending a trust relation update message to a central trust web agent, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a trust web.

11. (Original) The method of claim 10 further comprising:

receiving at the first certificate authority from the central trust web agent a set of inter-CA trust path indicators that represent whether a trust path exists between the first certificate authority and other certificate authorities in the trust web; and

receiving at the first certificate authority from the central trust web agent a set of shortest trust paths between the first certificate authority and other certificate authorities in the trust web.

12. (Original) The method of claim 11 further comprising:

initiating a secure communication with a requester;

receiving a digital certificate for the requester; and

validating the digital certificate in accordance with an inter-CA trust path indicator and/or a shortest trust path.

13. (Original) The method of claim 12 wherein the digital certificate is formatted according to X.509 standards.

14. (Original) An apparatus for processing information related to operations of certificate authorities within a data processing system, the apparatus comprising:

means for establishing at a first certificate authority (CA) a trust relation with a second certificate authority; and

means for sending a trust relation update message to a central trust web agent, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a trust web.

15. (Original) The apparatus of claim 14 further comprising:

means for receiving at the first certificate authority from the central trust web agent a set of inter-CA trust path indicators that represent whether a trust path exists between the first certificate authority and other certificate authorities in the trust web; and

means for receiving at the first certificate authority from the central trust web agent a set of shortest trust paths between the first certificate authority and other certificate authorities in the trust web.

16. (Original) The apparatus of claim 14 further comprising:

means for initiating a secure communication with a requester;

means for receiving a digital certificate for the requester; and

means for validating the digital certificate in accordance with an inter-CA trust path indicator and/or a shortest trust path.

17. (Original) The apparatus of claim 16 wherein the digital certificate is formatted according to X.509 standards

18. (Original) A computer program product in a computer-readable medium for use in a data processing system for processing information related to operations of certificate authorities within a data processing system, the computer program product comprising:

instructions for establishing at a first certificate authority (CA) a trust relation with a second certificate authority; and

instructions for sending a trust relation update message to a central trust web agent, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a trust web.

19. (Original) The computer program product of claim 18 further comprising:

instructions for receiving at the first certificate authority from the central trust web agent a set of inter-CA trust path indicators that represent whether a trust path exists between the first certificate authority and other certificate authorities in the trust web; and

instructions for receiving at the first certificate authority from the central trust web agent a set of shortest trust paths between the first certificate authority and other certificate authorities in the trust web.

20. (Original) The computer program product of claim 18 further comprising:

instructions for initiating a secure communication with a requester;

instructions for receiving a digital certificate for the requester; and

instructions for validating the digital certificate in accordance with an inter-CA trust path indicator and/or a shortest trust path.

21. (Original) The computer program product of claim 20 wherein the digital certificate is formatted according to X.509 standards

22. (Original) A method for operating certificate authorities within a data processing system, the method comprising:

receiving at a central trust web agent from a certificate authority (CA) a trust relation update message, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a trust web, and wherein the trust relation update message indicates a change in a set of trust relations for the certificate authority; and

modifying a set of trust relations for the set of certificate authorities within the trust web based on an indicated request in the trust relation update message.

23. (Original) The method of claim 22 further comprising:

sending to the certificate authority from the central trust web agent a set of inter-CA trust path indicators that represent whether a trust path exists between the certificate authority and other certificate authorities in the trust web; and

sending to the certificate authority from the central trust web agent a set of shortest trust paths between the certificate authority and other certificate authorities in the trust web.

24. (Original) The method of claim 22 further comprising:

representing the set of trust relations in an adjacency matrix, wherein a cell in the adjacency matrix corresponds to a pair of certificate authorities;

performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities; and

performing an all-pairs-shortest-paths computation on the adjacency matrix to generate multiple sets of shortest trust paths between the certificate authorities.

25. (Original) An apparatus for processing information related to operations of certificate authorities within a data processing system, the apparatus comprising:

means for receiving at a central trust web agent from a certificate authority (CA) a trust relation update message, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a trust web, and wherein the trust relation update message indicates a change in a set of trust relations for the certificate authority; and

means for modifying a set of trust relations for the set of certificate authorities within the trust web based on an indicated request in the trust relation update message.

26. (Original) The apparatus of claim 25 further comprising:

means for sending to the certificate authority from the central trust web agent a set of inter-CA trust path indicators that represent whether a trust path exists between the certificate authority and other certificate authorities in the trust web; and

means for sending to the certificate authority from the central trust web agent a set of shortest trust paths between the certificate authority and other certificate authorities in the trust web

27. (Original) The apparatus of claim 25 further comprising:

means for representing the set of trust relations in an adjacency matrix, wherein a cell in the adjacency matrix corresponds to a pair of certificate authorities;

means for performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities; and

means for performing an all-pairs-shortest-paths computation on the adjacency matrix to generate multiple sets of shortest trust paths between the certificate authorities.

28. (Original) A computer program product in a computer-readable medium for use in a data processing system for processing information related to operations of certificate authorities within a data processing system, the computer program product comprising:

instructions for receiving at a central trust web agent from a certificate authority (CA) a trust relation update message, wherein the central trust web agent processes trust relation information for a set of certificate authorities within a trust web, and wherein the trust relation update message indicates a change in a set of trust relations for the certificate authority; and

instructions for modifying a set of trust relations for the set of certificate authorities within the trust web based on an indicated request in the trust relation update message.

29. (Original) The computer program product of claim 28 further comprising:

instructions for sending to the certificate authority from the central trust web agent a set of inter-CA trust path indicators that represent whether a trust path exists between the certificate authority and other certificate authorities in the trust web; and

instructions for sending to the certificate authority from the central trust web agent a set of shortest trust paths between the certificate authority and other certificate authorities in the trust web.

30. (Original) The computer program product of claim 28 further comprising:

instructions for representing the set of trust relations in an adjacency matrix, wherein a cell in the adjacency matrix corresponds to a pair of certificate authorities;

instructions for performing a transitive closure computation on the adjacency matrix to generate a set of inter-CA trust path indicators that represent whether a trust path exists between a pair of certificate authorities; and

instructions for performing an all-pairs-shortest-paths computation on the adjacency matrix to generate multiple sets of shortest trust paths between the certificate authorities.

31-36. (Canceled)